

ESPECIFICACIÓN DE PERFIL ZIGBEE PARA MONITOREO Y CONTROL DE PLANTAS INDUSTRIALES

SERGIO SERNA¹

Resumen

ZigBee es una tecnología inalámbrica de corto alcance, bajo consumo de potencia, baja transmisión de datos y alta seguridad de uso creciente en diversas aplicaciones. Los perfiles ZigBee permiten utilizar esta tecnología para sistemas específicos, de tal forma que se desarrollen aplicaciones distribuidas que se puedan desplegar fácilmente en cualquier dominio de interés. Este artículo, presenta la especificación formal de un perfil ZigBee para monitoreo y control de plantas industriales, el cual se implementa utilizando herramientas de última generación para microcontroladores de 8 bits. Se despliega una red en un ambiente controlado que utiliza el perfil especificado, lográndose facilidad en el desarrollo de aplicaciones para redes inalámbricas de área personal, así como alta confiabilidad en la transmisión de datos.

Palabras clave:

ZigBee, PAN, IEEE 802.15.4, perfil, monitoreo y control

Abstract

The use of ZigBee wireless technology in many applications is currently increasing due its short-range, low power consumption,

1 MSc(c) Ingeniería de Sistemas
Académico Investigador
INSTITUTO TECNOLÓGICO METROPOLITANO
sergioserna@itm.edu.co

Fecha de recepción: 3 de marzo de 2009

Fecha de aceptación: 4 de agosto de 2009

low data transmission and high security. ZigBee profiles allow use this technology for specific systems, in order to develop of distributed applications to be easily implemented in any domain of interest. This paper presents the formal specification of a ZigBee profile for monitoring and control of industrial plants, which is implemented by use of cutting-edge tools for 8-bits microcontrollers. The network is deployed in a controlled environment using the specified profile, achieving the development of personal applications and higher confiability in the data transmission.

Keywords

ZigBee, PAN, IEEE 802.15.4, profile, monitoring and control

1. INTRODUCCIÓN

El crecimiento de las redes de área personal (PAN – Personal Area Network) se ha visto impulsado por el auge de los dispositivos que soportan tecnologías para este tipo de redes, como PCs, celulares y PDAs, y por el interés de los usuarios que cada vez quieren tener el control de más y más elementos, sin el uso de cables. Bluetooth lleva la vanguardia como tecnología de comunicaciones inalámbricas para redes PAN (SIG, 2006; Thomson, 2007). Sin embargo, con tanta diversidad de aplicaciones, algunas características de Bluetooth no son suficientes para satisfacer los requerimientos de los nuevos usuarios (Liu et al., 2006; Rashid & Yusoff, 2006), es por esta razón que emergen más tecnologías.

Dentro de las múltiples posibles aplicaciones en las que las redes PAN juegan un papel crucial, están el monitoreo remoto y no obstrusivo de parámetros fisiológicos de pacientes (Kim et al., 2007; Kulkarni & Öztürk, 2007), el monitoreo de condiciones climáticas, que permite en zonas de cultivo, determinar el nivel de químicos y riego necesario para cierto sembrado, o evitar desastres ambientales como inundaciones e incendios (Steere et al., 2000), el monitoreo de especies animales con el fin de determinar su población, posición y hábitat (Szewczyk et al., 2004), y así entre otras.

Una de las aplicaciones de más amplio crecimiento es el monitoreo y control inalámbrico de plantas industriales (Sasaki & Ueda, 2007; Cao, et al., 2008; Li et al., 2008; Zhao et al., 2008), para ello, es necesario utilizar a los sensores y actuadores como nodos dentro de una red (Gauger et al., 2008).

Un sensor inalámbrico consta de uno o varios transductores, los cuales capturan la información del ambiente, y un transmisor/receptor de radio frecuencia (RF). Un conjunto de este tipo de sensores inalámbricos pueden organizarse entre sí para formar una red ad hoc, y a su vez interactuar con un nodo maestro que reciba comandos de un operador humano u otro sistema (Akyildiz et al., 2002; Akkaya & Younis, 2005), de esta forma, es posible asignar

tareas de procesamiento, captura o transferencia de datos a cada sensor de la red.

En los últimos años, ha emergido una tecnología inalámbrica de baja velocidad de transmisión de datos, que ofrece confiabilidad, robustez, escalabilidad, seguridad y bajo consumo de potencia. Esta tecnología está respaldada por ZigBee Alliance, una organización que reúne a más de 300 compañías y que se encarga de definir las capas de red y aplicación de la pila de protocolos ZigBee (ZigBee Alliance, 2008).

ZigBee está construido sobre el estándar IEEE 802.15.4 el cual define las capas física (PHY – Physical) y de control de acceso al medio (MAC – Medium Access Control) (IEEE, 2003). La capa PHY soporta tres bandas de frecuencia: 2450 MHz (16 canales), 915 MHz (10 canales) y 868 MHz (1 canal); utiliza el modo de acceso de espectro disperso por secuencia directa (Direct Sequence Spread Spectrum – DSSS); la banda de 2450 MHz emplea modulación por desplazamiento de fase en cuadratura con offset (Offset Quadrature Phase Shift Keying – OQPSK), mientras que las otras dos bandas utilizan modulación por desplazamiento de fase binaria (Binary Phase Shift Keying – BPSK). La capa PHY soporta funcionalidades para la selección del canal, estimar la calidad del enlace, medir la energía detectada y otras más.

La capa MAC define dos tipos de nodos: los dispositivos de funciones reducidas (Reduced Function Devices -- RFDs) y los dispositivos con funciones completas (Full Function Devices -- FFDs). Un FFD puede actuar como coordinador de la red o dispositivo final. En su papel de coordinar, el FFD puede enviar balizas, y sincronizar y administrar la red como un todo. Los RFDs pueden actuar únicamente como dispositivos finales, regularmente equipados con sensores/actuadores, y comunicarse con un solo coordinador.

El estándar ZigBee (ZigBee Alliance, 2008) define las capas superiores de la pila de protocolos. La capa de red (Network Layer – NWK) se encarga de organizar y proveer enrutamiento en

una red de múltiples saltos, y soporta topologías en estrella, malla y árbol (véase Figura 1).

Son tres los tipos de dispositivos con los que cuenta ZigBee: coordinador y enrutador (corresponden a un FDD IEEE), y dispositivo final (corresponde a un RFD o FDD IEEE). El coordinador se encarga de incluir/excluir nodos a/de la red, la inicia y elige algunos parámetros claves cuando la topología de la red es malla o árbol. Igualmente maneja el identificador PAN único y el canal de operación. Una vez establecida la red, este dispositivo puede operar como un enrutador.

Los enrutadores son dispositivos opcionales en la red, y tienen la capacidad de extenderla. Una vez en la red, pueden asociarse con otros enrutadores o con el coordinador. Si la topología es árbol, se mueven de manera jerárquica mensajes de control y datos como estrategia de enrutamiento.

Los dispositivos finales trabajan en conjunto con los coordinadores y enrutadores. Se optimizan para consumir muy poca potencia y no cumplen labores de asociación ni de enrutamiento.

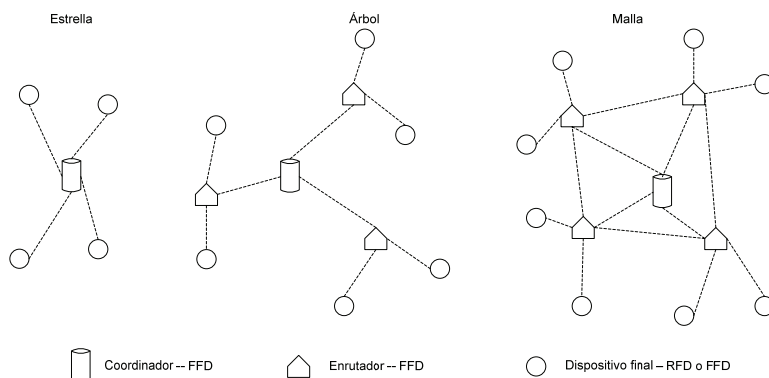


FIGURA 1. TOPOLOGÍAS DE RED EN ZIGBEE

La capa de aplicación (Application Layer – APL) provee un marco de trabajo para el desarrollo de aplicaciones distribuidas (véase figura 2). La APL los objetos de aplicación (Application

Objects – APOs), los objetos para dispositivos ZigBee (ZigBee Device Objects – ZDO), y la subcapa de aplicación (Application Sub Layer – APS).

Un APO es una pieza de software que controla una unidad hardware (transductor, conmutador, lámpara) disponible en el dispositivo. Cada APO tiene un identificador único. Un ZDO es un objeto especial que ofrece servicios a los APOs (descubrir dispositivos en la red y que servicios implementa, servicios de administración de seguridad, trabajo en red y comunicación). La APS proporciona servicios de transferencia de datos para los APOs y los ZDO.

Una aplicación ZigBee debe ajustarse a un perfil ZigBee (ZigBee Alliance, 2007a). Un perfil define el formato de los mensajes y las reglas de interacción entre los APOs, los cuales juntos forman una aplicación distribuida. De esta forma, es posible que diferentes desarrolladores, en forma independiente, construyan dispositivos ZigBee que puedan interactuar entre ellos. Cada APO encapsula un conjunto de atributos, y proporciona funcionalidad para establecer o recuperar valores de estos atributos, así como para notificar cuando un atributo cambia. En el contexto de un perfil, un conjunto de atributos relacionados se denomina “cluster”, y se le asigna un identificador numérico (ZigBee Alliance, 2007b).

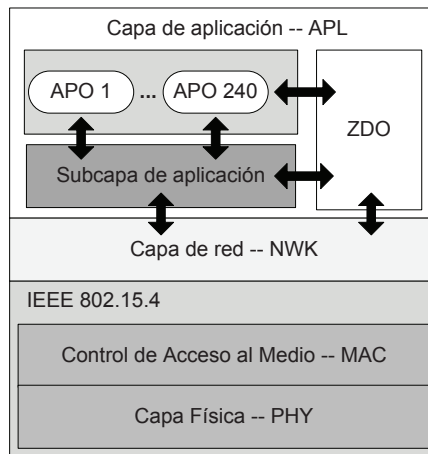


FIGURA 2. PILA DE PROTOCOLOS Y ARQUITECTURA EN CAPAS ZigBEE

Este artículo definirá un perfil que permitirá la utilización de la tecnología ZigBee en sistemas de monitoreo y control de procesos en laboratorios académicos, de esta forma es posible ocuparse del análisis de los resultados de la evaluación de los sistemas, y no de la adquisición y transmisión de la información. Se logrará igualmente flexibilidad en el tipo de plantas a monitorear y controlar, y movilidad en los actuadores y sensores, gracias a la tecnología inalámbrica utilizada.

Este artículo se estructura como sigue: la próxima sección relaciona los métodos y las herramientas utilizadas en la investigación, luego se caracterizan los perfiles, y se propone el de monitoreo y control de plantas industriales. Posteriormente se describe la aplicación utilizada para probar el perfil, describiendo la planta, el sistema embebido y la aplicación para el usuario final. La última sección destaca las conclusiones e igualmente direcciona los trabajos futuros.

2. DESCRIPCIÓN DEL SISTEMA

Un sistema de control típico sobre el cual se puede implementar el perfil especificado se muestra en la Figura 3. La utilidad de estos sistemas consiste en el hecho de que incluyen una variada instrumentación, la cual sirve para la construcción del prototipo inicial. El sistema de la figura consiste de dos tanques, a los cuales se les mide el nivel por medio de sensores de presión y se controla el llenado con electroválvulas. El sensor de flujo entrega la señal de realimentación al controlador y la bomba regula el flujo.

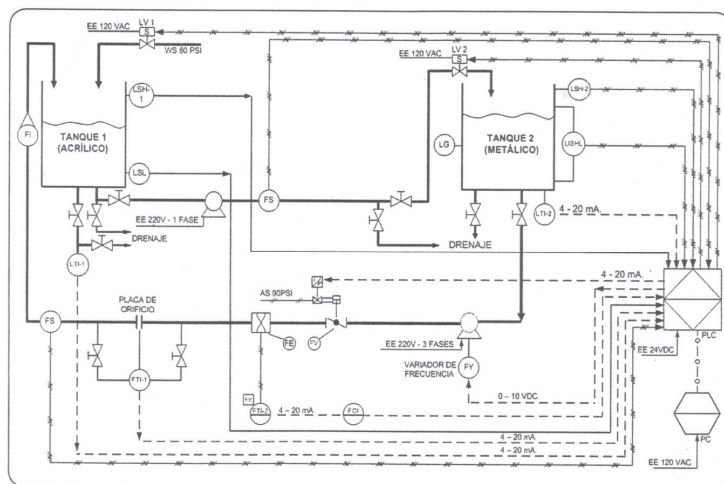


FIGURA 3. DIAGRAMA DE LA PLANTA DE CAUDAL

Luego de identificar la instrumentación de interés, se debe pasar a la identificación de los cluster existentes dentro del ZCL (ZigBee Alliance, 2007b), de esta manera, es posible reutilizar la especificación previamente establecida y que se encuentra consignada en el estándar. Posteriormente, se definen y especifican los cluster que no se encuentran en el ZCL de acuerdo al formato predeterminado allí. Esta especificación son los APOs de este desarrollo, los cuales se construyen sobre la pila de protocolos ZigBee.

La pila de protocolos ZigBee sobre la que se trabajó fue Freescale BeeStack™ (Freescale Semiconductor, 2007), la cual viene con el software BeeKit™ (Freescale Semiconductor, 2008a) y contiene la funcionalidad completa ZigBee. Se incluye en esta pila las capas PHY y MAC, y la implementación que Freescale hace de las capas NWK y APL.

Utilizando plantillas de BeeKit y exportándolas al CodeWarrior (Freescale Semiconductor, 2008b), se logra tener nodos RFD o FFD sobre microcontroladores de 8 bits. Esta pila ZigBee completa, más la funcionalidad adicional desarrollada para la aplicación,

se descarga sobre el chip MC13213 (System In Package – SIP), el cual integra el microcontrolador MC9S08GT con el transceiver MC1320x en un solo empaquetado LGA (Land Grid Array) de 9x9 mm (Freescalse Semiconductor, 2008c; 2008d). Este SIP es ideal para aplicaciones de monitoreo y control en redes de sensores inalámbricos en malla.

Con una aplicación en LabVIEW™ (National Instruments Corporation, 2008), se visualiza toda la red ZigBee, tanto los nodos finales como el nodo coordinador.

3. ESPECIFICACIÓN DEL PERFIL

Los perfiles son acuerdos que se establecen con el fin de que se facilite el intercambio de mensajes para un tipo de aplicación específica. Para esto es necesario tener un formato de mensaje, de tal manera que se puedan enviar comandos, se puedan requerir o transmitir datos, y demás.

Existen dos tipos de perfiles ZigBee: públicos y específicos. Aparte del identificador del perfil que el diseñador debe solicitar a la alianza ZigBee, es necesario definir la descripción de los dispositivos y la identificación de los clusters.

a. Definición de los dispositivos

El perfil permite manejar los dispositivos propios de un proceso de control de nivel y flujo en un sistema de tanques, pero igualmente es posible extenderlo a otro tipo de procesos. Los dispositivos se describen en la Tabla 1, los cuales están organizados de acuerdo a si son actuadores o sensores/detectores.

La asignación de direcciones se distribuye de manera arbitraria, dejando lugar para la definición de otros dispositivos en el futuro. Por ejemplo, dependiendo del proceso, sería deseable tener en cuenta detectores de movimiento, sensores de posición lineal, sensores de posición angular, acelerómetros, sensores de deformación, detectores de obstáculos, sensores de PH, sensores de CO₂, entre otros.

No es necesario que se implementen todos los dispositivos para realizar las pruebas, sólo es suficiente con elegir los necesarios para el sistema específico. Este perfil no es muy extenso, pero la idea es que sea lo suficientemente amplio para que en su primera versión se puedan realizar las actividades más comunes del tipo de sistemas para el cual fue propuesto.

TABLA 1. DISPOSITIVOS ESPECÍFICOS PARA EL PERFIL DE MONITOREO Y CONTROL

Dispositivo	ID
válvula on/off	0x0001
válvula proporcional	0x0002
bomba on/off	0x0003
bomba proporcional	0x0004
Termostato	0x0005
Ventilador	0x0006
Reservado	0x0007 – 0x00FF
sensor de nivel	0x0101
sensor de distancia	0x0102
sensor de flujo	0x0103
sensor de presión	0x0104
sensor de temperatura	0x0105
sensor de humedad relativa	0x0106
sensor de intensidad lumínica	0x0107
detector de nivel	0x0108
detector de flujo	0x0109
detector de inclinación	0x010A
Reservado	0x010B – 0x01FF

b. Identificación de los clusters

La librería de cluster ZigBee (ZCL – ZigBee Cluster Library), es un repositorio de atributos, los cuales definen el estado de los dispositivos, y un conjunto de comandos, los cuales permiten la comunicación entre dos dispositivos (ZigBee Alliance, 2007b). Los dispositivos deben implementar las interfaces del lado del cliente y del lado del servidor. La entidad que almacena los atributos del cluster es el servidor, la que manipula dichos atributos es el cliente. Sin embargo, si se requiere, los atributos pueden estar presentes en el lado del cliente.

Al construirse un nuevo perfil, es necesario especificar cuales clusters requieren cada uno de los dispositivos, de esta forma, inicialmente es necesario buscar dentro de la ZCL con el fin de extraer de allí la funcionalidad requerida por el perfil. Sin embargo, es posible definir nuevos clusters, para lo cual debe solicitarse su inclusión en la ZCL. La ZCL tiene dos componentes que la definen: un conjunto de elementos que se aplica a toda la librería, tales como la estructura de la trama, el acceso a los atributos de los comandos y los tipos de datos, y un número de conjuntos de cluster que son utilizados en aplicaciones de un dominio específico.

De acuerdo a los dispositivos referidos en la Tabla 1, los clusters utilizados en este perfil se muestran en la Tabla 2. Todos los clusters, menos *Distance measurement*, *Controller* y *Plant* están definidos en (ZigBee Alliance, 2007b), por lo tanto es necesario incluirlos en la ZCL luego de especificarlos detalladamente.

TABLA 2. CLÚSTER UTILIZADOS EN EL PERFIL

Dispositivo	ID
Basic	0x0000
Identify	0x0003
on/off	0x0006
on/off switch configuration	0x0007
Level Control	0x0008
Alarms	0x0009
Illuminance measurement	0x0400
Illuminance level sensing	0x0401
Temperature measurement	0x0402
Pressure measurement	0x0403
Flow measurement	0x0404
Relative humidity measurement	0x0405
Pump Configuration and Control	0x0200
Thermostat	0x0201
Fan Control	0x0202
Distance measurement	0x0700
Controller	0x0701
Plant	0x0702

a. Clúster Distance measurement

El lado del servidor de este cluster es una interfaz para la medición de distancia, incluye configuración y notificación del valor medido. La definición de atributos sigue, por conveniencia, el formato establecido en (ZigBee Alliance, 2007b), aquí el identificador del atributo es un número de 16 bits, donde los 12 bits de mayor peso indican el conjunto de atributos y los 4 bits restantes hablan de uno de 16 posibles atributos dentro del conjunto.

El lado del servidor tiene definidos los atributos para la medición de distancia, y las operaciones o comandos que debe ejecutar a las solicitudes del cliente. El lado del cliente no tiene atributos pero si operaciones o comandos, los cuales se utilizan para acceder al servidor con el fin de establecer o leer sus atributos (véase Figura 4).

<table><tr><th>DistanceMeasureClusterServer</th></tr><tr><td>-measuredValue : unsigned int</td></tr><tr><td>-minMeasuredValue : unsigned int</td></tr><tr><td>-maxMeasuredValue : unsigned int</td></tr><tr><td>-tolerance : unsigned int</td></tr><tr><td>+ReadAttributesResponse() : unsigned int*</td></tr><tr><td>+WriteAttributesResponse() : unsigned int*</td></tr><tr><td>+Discover () : attributes*</td></tr></table>	DistanceMeasureClusterServer	-measuredValue : unsigned int	-minMeasuredValue : unsigned int	-maxMeasuredValue : unsigned int	-tolerance : unsigned int	+ReadAttributesResponse() : unsigned int*	+WriteAttributesResponse() : unsigned int*	+Discover () : attributes*	<table><tr><th>DistanceMeasureClusterClient</th></tr><tr><td></td></tr><tr><td>+ReadAttributes()</td></tr><tr><td>+WriteAttributes()</td></tr><tr><td>+WriteAttributesUndivided()</td></tr><tr><td>+WriteAttributesNoResponse()</td></tr><tr><td>+DiscoverAttributes()</td></tr></table>	DistanceMeasureClusterClient		+ReadAttributes()	+WriteAttributes()	+WriteAttributesUndivided()	+WriteAttributesNoResponse()	+DiscoverAttributes()
DistanceMeasureClusterServer																
-measuredValue : unsigned int																
-minMeasuredValue : unsigned int																
-maxMeasuredValue : unsigned int																
-tolerance : unsigned int																
+ReadAttributesResponse() : unsigned int*																
+WriteAttributesResponse() : unsigned int*																
+Discover () : attributes*																
DistanceMeasureClusterClient																
+ReadAttributes()																
+WriteAttributes()																
+WriteAttributesUndivided()																
+WriteAttributesNoResponse()																
+DiscoverAttributes()																
<table><tr><th>ControllerClusterServer</th></tr><tr><td>-proportional : unsigned int</td></tr><tr><td>-integral : unsigned int</td></tr><tr><td>-derivative : unsigned int</td></tr><tr><td>+ReadAttributesResponse() : unsigned int*</td></tr><tr><td>+WriteAttributesResponse() : unsigned int*</td></tr><tr><td>+DiscoverAttributesResponse () : attributes*</td></tr></table>	ControllerClusterServer	-proportional : unsigned int	-integral : unsigned int	-derivative : unsigned int	+ReadAttributesResponse() : unsigned int*	+WriteAttributesResponse() : unsigned int*	+DiscoverAttributesResponse () : attributes*	<table><tr><th>ControllerClusterClient</th></tr><tr><td></td></tr><tr><td>+ReadAttributes() : decimal</td></tr><tr><td>+WriteAttributes()</td></tr><tr><td>+WriteAttributesUndivided()</td></tr><tr><td>+WriteAttributesNoResponse()</td></tr><tr><td>+DiscoverAttributes()</td></tr></table>	ControllerClusterClient		+ReadAttributes() : decimal	+WriteAttributes()	+WriteAttributesUndivided()	+WriteAttributesNoResponse()	+DiscoverAttributes()	
ControllerClusterServer																
-proportional : unsigned int																
-integral : unsigned int																
-derivative : unsigned int																
+ReadAttributesResponse() : unsigned int*																
+WriteAttributesResponse() : unsigned int*																
+DiscoverAttributesResponse () : attributes*																
ControllerClusterClient																
+ReadAttributes() : decimal																
+WriteAttributes()																
+WriteAttributesUndivided()																
+WriteAttributesNoResponse()																
+DiscoverAttributes()																
<table><tr><th>PlantClusterServer</th></tr><tr><td>-order : unsigned char</td></tr><tr><td>-timeConstant : unsigned int</td></tr><tr><td>-naturalFrequency : unsigned int</td></tr><tr><td>-damping : unsigned int</td></tr><tr><td>+ReadAttributesResponse() : unsigned int*</td></tr><tr><td>+WriteAttributesResponse() : unsigned int*</td></tr><tr><td>+DiscoverAttributesResponse () : attributes*</td></tr></table>	PlantClusterServer	-order : unsigned char	-timeConstant : unsigned int	-naturalFrequency : unsigned int	-damping : unsigned int	+ReadAttributesResponse() : unsigned int*	+WriteAttributesResponse() : unsigned int*	+DiscoverAttributesResponse () : attributes*	<table><tr><th>PlantClusterClient</th></tr><tr><td></td></tr><tr><td>+ReadAttributes()</td></tr><tr><td>+WriteAttributes()</td></tr><tr><td>+WriteAttributesUndivided ()</td></tr><tr><td>+WriteAttributesNoResponse()</td></tr><tr><td>+DiscoverAttributes()</td></tr></table>	PlantClusterClient		+ReadAttributes()	+WriteAttributes()	+WriteAttributesUndivided ()	+WriteAttributesNoResponse()	+DiscoverAttributes()
PlantClusterServer																
-order : unsigned char																
-timeConstant : unsigned int																
-naturalFrequency : unsigned int																
-damping : unsigned int																
+ReadAttributesResponse() : unsigned int*																
+WriteAttributesResponse() : unsigned int*																
+DiscoverAttributesResponse () : attributes*																
PlantClusterClient																
+ReadAttributes()																
+WriteAttributes()																
+WriteAttributesUndivided ()																
+WriteAttributesNoResponse()																
+DiscoverAttributes()																

FIGURA 4. DEFINICIÓN DE LOS CLÚSTER DEL PERFIL

El atributo *measuredValue* representa la distancia en *cm*, con una resolución de *0,1cm*. Como se representa con un entero no signado de 16 bits, el rango de posibles valores será, $0 \leq distance \leq 6553,4$. El valor *0xFFFF* indica que la distancia medida es inválida.

Los atributos *minMeasuredValue* y *maxMeasuredValue* indican el mínimo y máximo valor que se puede obtener en *measuredValue*. De esta forma es posible establecer el rango del sensor con estos dos atributos. Asignar a estos atributos el valor *0xFFFF* indica que el límite no está establecido. Claramente se debe cumplir que $maxMeasuredValue \leq minMeasuredValue$.

El atributo *tolerance* indica la magnitud del posible error asociado con el atributo *measuredValue*, lo que quiere decir que el valor real de la distancia se encuentra entre (*measuredValue - tolerance*) y (*measuredValue + tolerance*).

En el cliente se definen cinco comandos, uno para leer los atributos, *ReadAttributes*, otro para escribir un atributo esperando respuesta, *WriteAttributes*, otro más para escribir un grupo de atributos como una operación atómica esperando respuesta, *WriteAttributesUndivided*, uno más para escribir un atributo sin esperar respuesta del cliente, *WriteAttributesNoResponse* y por último un comando para solicitarle al servidor la información acerca de que atributos posee y de qué tipo, *DiscoverAttributes*. Estos cinco comandos en el cliente sólo requieren tres comandos en el servidor para ser atendidos. *ReadAttributesResponse* es la respuesta del servidor a *ReadAttributes*, entregando el valor de los atributos solicitados por el cliente. *WriteAttributesResponse* es la respuesta del servidor a *WriteAttributes* o *WriteAttributesUndivided*, devolviendo el valor en el que quedaron los atributos luego de la actualización, y por último *DiscoverAttributesResponses* retorna al cliente los atributos del servidor junto con su tipo, este es el comando respuesta a *DiscoverAttributes*.

b. Clúster Controller

El lado del servidor de este cluster es una interfaz para configurar un controlador clásico del tipo PID (Kuo, 2006),

incluye configuración y notificación de los parámetros actuales.

El lado del servidor tiene definidos los atributos para sintonizar el controlador, y las operaciones o comandos que debe ejecutar a las solicitudes del cliente. El lado del cliente no tiene atributos pero si operaciones o comandos, los cuales se utilizan para acceder al servidor con el fin de establecer o leer sus atributos (véase Figura 4).

Los parámetros del controlador incluyen dos decimales de resolución, lo que establece su rango entre 0 y 655,34, ya que el último valor indica atributo no válido. Si se establecen apropiadamente estos valores, es posible obtener controladores P, PI, PID o cualquier otra combinación utilizada en técnicas de control clásicas.

c. Clúster Plant

El lado del servidor de este cluster es una interfaz para configurar una planta de primer o segundo orden (Kuo, 2006), incluye configuración y notificación de los parámetros actuales.

El lado del servidor tiene definidos los atributos que identifican la planta, y las operaciones o comandos que debe ejecutar a las solicitudes del cliente. El lado del cliente no tiene atributos pero si operaciones o comandos, los cuales se utilizan para acceder al servidor con el fin de establecer o leer sus atributos (véase Figura 4).

A excepción del atributo *order*, los parámetros de la planta incluyen dos decimales de resolución, lo que establece su rango entre 0 y 655,34, ya que el último valor, 655,35, indica atributo no válido. El atributo *order* permite establecer parámetros para plantas de cualquier orden, entre 1 y 254, pero esos parámetros no están aún definidos como atributos del cluster. Por lo tanto si el atributo *order* es 1, debe establecerse el atributo *timeConstant*, mientras que los atributos *naturalFrequency* y *damping* se llevan al valor no válido 0xFFFF. Cuando el atributo *order* se hace 2, los valores de los atributos *naturalFrequency* y *damping* son los válidos para la planta, o sea deben establecerse, mientras que *timeConstant* se lleva a valor no válido, 0xFFFF.

4. RESULTADOS

Para probar las bondades del perfil, se desplegó una red de 7 nodos, un coordinador y seis nodos finales, los cuales incluyen nodos sensores y actuadores. Las señales que se censaron provienen de convertidores de corriente a voltaje, los actuadores son dispositivos lumínicos tipo LEDs. La dinámica de la red, esto es, el ingreso y salida de nodos, es un proceso bastante simple realizado con facilidad por el coordinador. En un ambiente limpio de contaminación electromagnética y distancias cortas, no se detectó pérdida de información. Solicitudes de censado y activación de actuadores en forma “simultánea” se realizaron sin dificultad. La programación de la red fue bastante simple, ya que la trama del perfil se tenía implementada y sólo fue necesario ocuparse de la aplicación, y los procesos que esta conlleva.

Una dificultad que se presentó fue al momento de utilizar módems de diferentes fabricantes para armar la red, ya que a pesar de tener el mismo perfil implementado, la pila ZigBee parece no tener las mismas prestaciones en todos los diferentes casos.

5. CONCLUSIONES

La especificación de un perfil para monitoreo y control de plantas industriales, facilita la implementación de soluciones en este dominio de aplicación, ya que el desarrollador puede ocuparse exclusivamente de los problemas que trae consigo la aplicación, haciendo a un lado los problemas de comunicaciones inherentes a las WPAN.

El perfil de monitoreo y control de plantas industriales propuesto, se integra totalmente con el ZCL, de esta manera se mantienen los lineamientos establecidos por la ZigBee Alliance y así se satisface uno de los requisitos necesarios para lograr interoperabilidad entre diversos fabricantes. Sin embargo, debido a lo reciente del estándar y la tecnología, existen implementaciones de las pilas ZigBee que no parecen cumplir las especificaciones de la ZigBee Alliance al detalle, por esta

razón aún no se logran redes ZigBee operativas con módems de diferentes fabricantes.

Al definir el perfil, se logro establecer un lenguaje común para el intercambio de datos, y la rigurosidad en la definición de las acciones de procesamiento, trajo como consecuencia un sistema mucho más simple y confiable, ya que de esta manera el usuario del perfil sólo debe preocuparse por su aplicación, y no por la compatibilidad en el proceso de comunicación. Si bien no existe un perfil ZigBee para este tipo de sistemas, muchos de los cluster necesarios ya están definidos en la ZCL, por lo tanto el esfuerzo en nuevas especificaciones se reduce bastante.

Los estándares establecidos por la ZigBee Alliance y la IEEE para las WPAN no han logrado unificar completamente a los múltiples fabricantes de dispositivos ZigBee que existen en el medio, ya que cada uno tienen implementaciones propietarias de la pila ZigBee en alguna de sus capas, lo que evita la interoperabilidad.

Los ambientes limpios de ruido electromagnético son ideales para la comunicación con esta tecnología de corto alcance, ya que a pesar de trabajar en la banda ISM (Industrial, Scientific and Medical), su selección del canal, modulación y modo de acceso minimizan las posibilidades de interferencias.

La facilidad que da el estándar ZigBee de definir perfiles, facilita la extensión de la tecnología a prácticamente cualquier dominio de aplicación.

Con las múltiples herramientas de desarrollo, y el hardware integrado de microcontrolador y módem en un solo chip, el tiempo de desarrollo de aplicaciones profesionales de redes inalámbricas es bastante reducido.

El proyecto debe continuar con el despliegue de la red de sensores y actuadores en una planta industrial real o académica, donde las condiciones introduzcan otras variables que aún no han sido analizadas en este artículo, como es el ruido y las exigencias de desempeño de una planta de tiempo real.

RECONOCIMIENTOS

Este artículo es el resultado del proyecto *Red de Monitoreo y Control Utilizando Tecnología ZigBee*, que desarrolla el grupo de investigación en integración de soluciones con TICs – GIT, y está financiado por el INSTITUTO TECNOLÓGICO METROPOLITANO.

BIBLIOGRAFÍA

- Akkaya, K., & Younis, M. (2005). A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3 (3), 325-349.
- Akyildiz, I., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). A Survey on Sensor Networks. *Communications Magazine, IEEE*, 40 (8), 102 - 114.
- Cao, J., Qian, S., Hu, H., & Yang, S. (2008). Wireless monitoring and analysing system of dissolved gases in the transformer oil. *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on* (págs. 5152 - 5157). Chongqing, China: IEEE.
- Freescale Semiconductor. (2007). *Freescale BeeStack™ Software Reference Manual*. Denver, Colorado: Freescale Document BSSRM.
- Freescale Semiconductor. (2008a). *BeeKit Wireless Connectivity Toolkit User's Guide*. Denver, Colorado: Freescale Document BKWCTKUG.
- Freescale Semiconductor. (2008b). *CodeWarrior™ Development Studio for Microcontrollers V6.2*. Denver, Colorado: Freescale Document 950-00087.
- Freescale Semiconductor. (2008c). *MC13211/212/213 ZigBee™. Compliant Platform - 2.4 GHz Low Power Transceiver for the IEEE® 802.15.4 Standard plus Microcontroller Reference Manual*. Denver, Colorado: Freescale Document MC1321xRM.
- Freescale Semiconductor. (2008d). *MC1321x Evaluation Kit (EVK) Reference Manual*. Denver, Colorado: Freescale Document MC1321xEVKRM.
- Gauger, M., Minder, D., Marrón, P. J., Wacker, A., & Lachenmann, A. (2008). Prototyping sensor-actuator networks for home automation. *REALWSN '08: Proceedings of the workshop on Real-world wireless sensor networks* (págs. 56-60). Glasgow, Scotland: ACM SIGOPS.
- IEEE (2003). Institute of Electrical and Electronics Engineers, Inc. *IEEE std. 802.15.4 Wireless Medium Access Control (MAC) and Physical Layer*

- (PHY) *Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)*. New York: IEEE Press.
- Kim, B., Kim, Y., Lee, I., & You, I. (2007). Design and Implementation of a Ubiquitous ECG Monitoring System Using SIP and the Zigbee Network. *Future generation communication and networking (fgcn 2007)*, 2, 599 - 604.
- Kulkarni, P., & Öztürk, Y. (2007). Requirements and design spaces of mobile medical care. En ACM, *ACM SIGMOBILE Mobile Computing and Communications Review* (Vol. 11, págs. 12-30). New York, NY, USA.
- Kuo, B. (2006). *Sistemas de Control Digital*. México: Compañía Editorial Continental.
- Li, X., Fang, K., Gu, J., & Zhang, L. (2008). An Improved ZigBee Routing Strategy for Monitoring System. *Intelligent Networks and Intelligent Systems, 2008. ICINIS '08. First International Workshop on* (págs. 255 - 258). Wuhan, China: IEEE.
- Liu, Y., Yu, S., Chen, W., & Li, W. (2006). Wireless Communication Technology Based on Bluetooth and Its Application to a Manipulator. *2006 IEEE International Conference on Industrial Informatics* (págs. 1251 - 1256). Singapore: IEEE.
- National Instruments Corporation. (2008). *LabVIEW™ Help*. Austin, Texas: NI Document 371361E-01.
- Rashid, R., & Yusoff, R. (2006). Bluetooth Performance Analysis in Personal Area Network (PAN). *RF and Microwave Conference, 2006. RFM 2006. International* (págs. 393 - 397). Putrajaya, Malaysia: IEEE.
- Sasaki, H., & Ueda, K. (2007). Design and Deployment of Wireless Monitoring System for 4-20mA Current Loop Sensors. *Networked Sensing Systems, 2007. INSS '07. Fourth International Conference on* (págs. 73 - 76). Braunschweig, Germany: IEEE.
- SIG (2006). The Bluetooth Special Interest (SIG). *Installed base of more than one billion products gives consumers more than five billion ways to use the global wireless standard*. Technical Report.
- Steere, D., Baptista, A., McNamee, D., Pu, C., & Walpole, J. (2000). Research challenges in environmental observation and forecasting systems. *Proceedings of the 6th annual international conference on Mobile computing and networking* (págs. 292 - 299). Boston, Massachusetts, United States: ACM.
- Szewczyk, R., Mainwaring, A., Polastre, J., Anderson, J., & Culler, D. (2004). An Analysis of a Large Scale Habitat Monitoring Application. *Proceed-*

- ings 2nd International Conference on Embedded Networked Sensor Systems (SenSys 2004)* (págs. 214-226). Baltimore, MD, USA: ACM.
- Thomson, F. (2007). *Bluetooth enabled equipment shipments*. Technical Report, IMS Research.
- Trovao, J., Santos, F., Silva, M., & Jorge, H. (2008). A web-based monitoring approach for power systems in industrial plants. *Industrial Electronics, 2008. ISIE 2008. IEEE International Symposium on* (págs. 1769 - 1774). Cambridge, UK: IEEE.
- Zhao, Q., Gao, X., Wang, H., & Yang, Y. (2008). The application of oil and gas wells intelligent wireless monitoring system in oil field system. *IT in Medicine and Education, 2008. ITME 2008. IEEE International Symposium on* (págs. 906 - 910). Xiamen, China: IEEE.
- ZigBee Alliance. (2007a). *ZigBee Home Automation Public Application Profile*. San Ramon, CA: ZigBee Document 053520r25.
- ZigBee Alliance. (2007b). *ZigBee Cluster Library Specification*. San Ramon, CA: ZigBee Document 075123r01ZB.
- ZigBee Alliance. (2008). *ZigBee Specifications*. San Ramon, CA: ZigBee Document 053474r17.